Learning Geometric Phase Field Representations

Master‘s Thesis

Yannick Kees

Advisor: Prof. Dr. Martin Rumpf
Second Advisor: Prof. Dr. Tim Laux

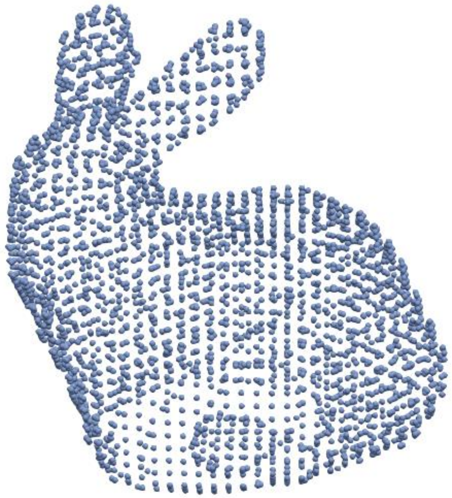
INITIAL PROBLEM

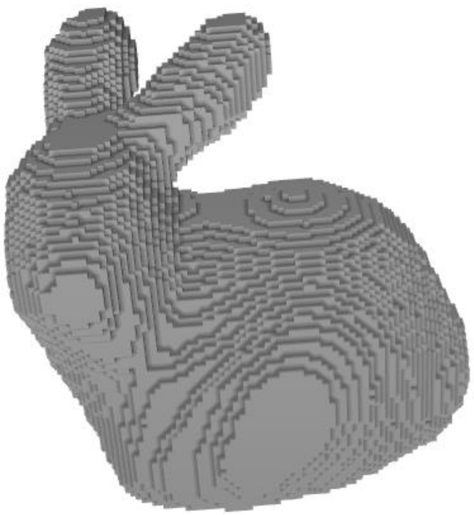
- best way to represent 3D data?

Storing geometric data explicitly

Point cloud

+ close to raw input f. Sensor data
- High memory requirements
- No information on topology



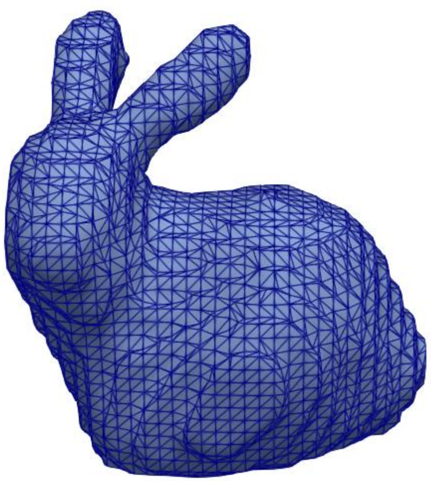


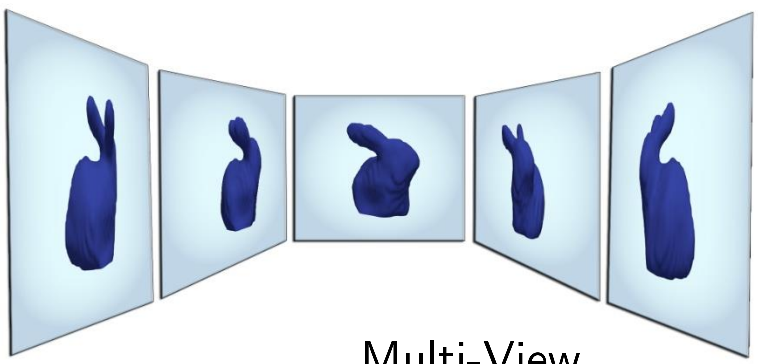
Voxeling

+ volumetric data (MRI)
+ math. Operations (+,*)
- #Cubes grows cubically

Meshes

+ straightforward to process
+ low memory requirements
- No solid objects





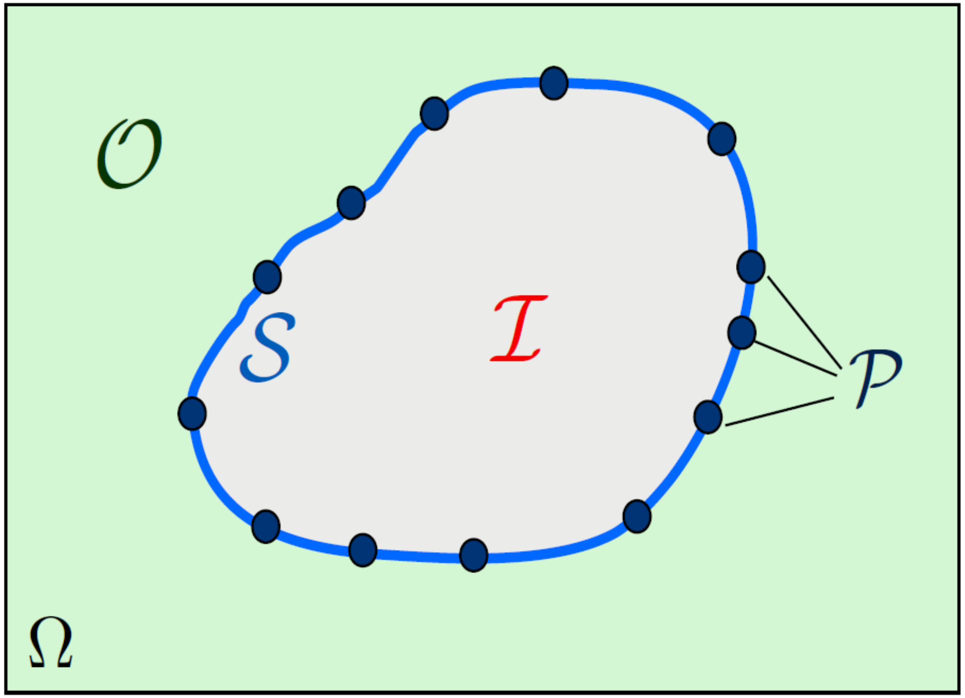
Multi-View

+ reduce to 2D problem
- #images needed

Implicit Representations

Let $\mathcal{P} \subset \mathcal{S} \subset \Omega$

Given $\mathcal{P}$, find an implicit representation for $\mathcal{S}$



Goal: Find $u : \Omega \to [-1, 1]$, such that:

$$\mathcal{S} = \{x \in \Omega : \; u(x) = 0\}$$

$$\mathcal{I} = \{x \in \Omega : \; u(x) < 0\}$$
$$\mathcal{O} = \{x \in \Omega : \; u(x) > 0\}$$

- ill posed

- Degree of freedom away from surface

4 Major types of Implicit functions

1. **Indicator functions.**

$$u_{IF}(x) = \begin{cases} 0 & x \in \mathcal{S} \\ 1 & x \in \mathcal{O} \cup \mathcal{I} \end{cases}$$

2. **Distance functions.**

$$u_{DF}(x) = d(x, \mathcal{S}) = \inf_{y \in \mathcal{S}} d(x, \mathcal{S}) = \inf_{y \in \mathcal{S}} \|x - y\|$$

$$\underbrace{\text{From now on, only Euclidean metric}}_{\text{Don't distinguish interior or exterior — Good for open surfaces}}$$

4 Major types of Implicit functions (cont.)

3. Occupancy function.

$$u_{OF}(x) = \begin{cases} 0 & x \in \mathcal{S} \\ 1 & x \in \mathcal{O} \\ -1 & x \in \mathcal{I} \end{cases}$$

- More general version of indicator functions

4. Signed distance functions (SDF).

$$u_{SDF}(x) = u_{OF}(x)u_{DF}(x)$$

- Gradient equals surface normal

distinguish interior or exterior — Good for closed surfaces

Implicit Neural Networks

Recently it got popular to choose u to be a neural network

[*Deep SDF: Learning Continuous Signed Distance Functions for Shape Representation*, Park et al., 2019]

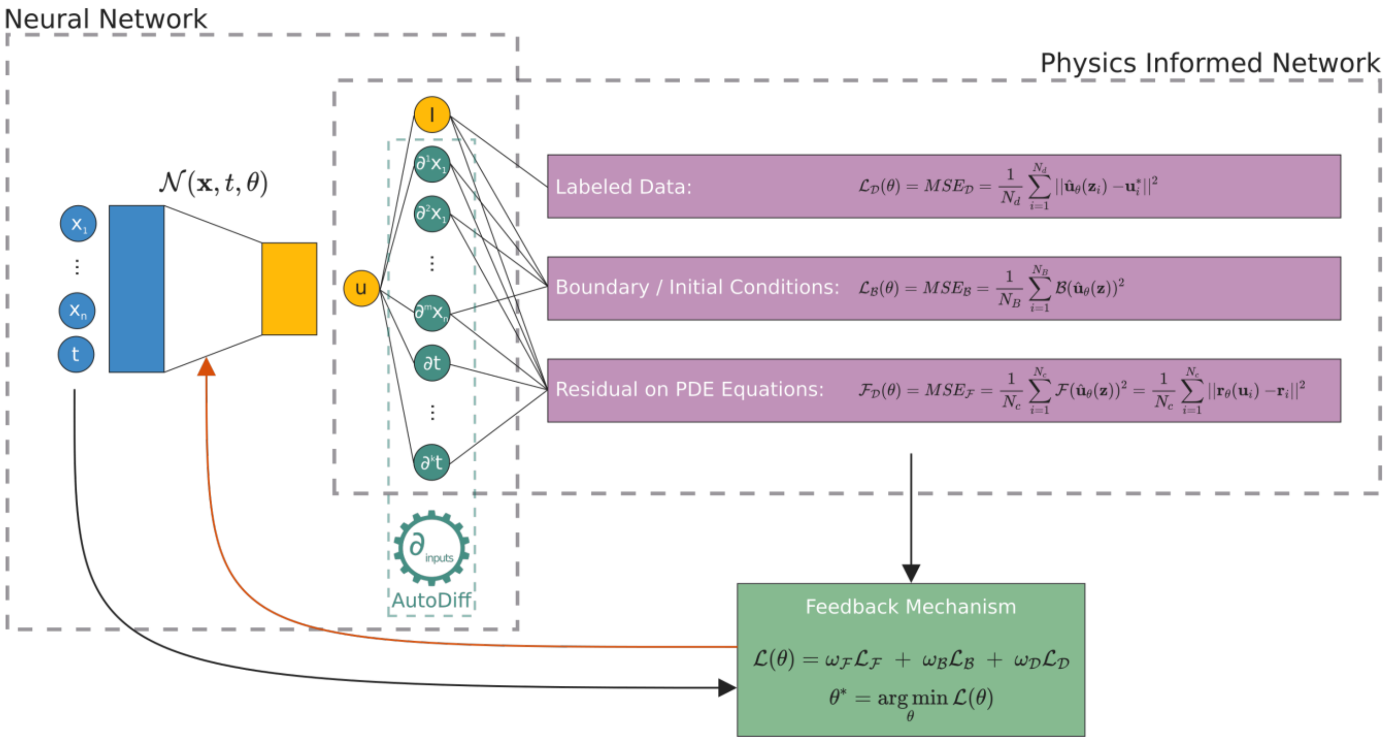
$$\mathcal{S} = \{x \in \Omega : \; u_\theta(x) = 0\}$$

- u is a NN with parameter theta
- Spacial discretization not dependent on target domain
- Supervised learning learning problem

How do we train the Neural Network/What loss functional do we choose?
Choose a Partial differential operator as loss!

PINN

- Consider a PDE & NN
- Autodifferentiation for evaluation
- Loss: PDE Residual + boundary conditions + labled data
- NN adjust themselves to minimize loss



- Solved PDE using NN
- Variational problems
- Results PINN
- Consider physics

[*Scientific Machine Learning through Physics-Informed Neural Networks: Where we are and What's next*, Cuomo et al., 2022]

1. Goal: Surface reconstruction

Given a single point cloud $\mathcal{P}$, find parameters θ such that

$$\mathcal{S} = \{x \in \Omega : \ u_\theta(x) = 0\} \supset \mathcal{P}$$

- For training the network:

Find a loss functional $\mathcal{L}$ such that $\mathcal{S} = \{x \in \Omega : u_{\theta_*}(x) = 0\}$
where $\theta_* = \text{argmin}_\theta \mathcal{L}(u_\theta, \mathcal{P})$

Universal approximation & PINN: drop NN

Find a variational problem $\mathcal{L}$ such that
$\mathcal{S} = \{x \in \Omega : u_*(x) = 0\}$ where $u_* = \text{argmin}_u \mathcal{L}(u, \mathcal{P})$

2. Goal: Shape space learning

Reconstruct the surfaces of multiple shapes with single network

$$\mathcal{S}_i = \{x \in \Omega : \ u_\theta(x, \mathcal{P}_i) = 0\}$$

Find a loss functional $\mathfrak{L}$ such that $\mathcal{S} = \{x \in \Omega : u_{\theta_*}(x, \mathcal{P}_i) = 0\}$ where $\theta_* = \text{argmin}_\theta \mathfrak{L}(u_\theta, \{\mathcal{P}_i\}_i)$

- Map $\mathcal{P}_i \mapsto z_i$
- Having solved SR with $\mathcal{L}$, then

$$\mathfrak{L} = \sum_i \mathcal{L}(u_\theta(\cdot, z_i), \mathcal{P}_i)$$

- enough to focus on SR

MODICA-MORTOLA-BASED

PHASE-Loss

[*Phase Transitions, Distance Functions, and Implicit Neural Representations*, Lipman, 2021]

Goals:

- **Proper occupancy.** $u \in BV(\Omega; \{-1, 1\})$
- **Zero reconstruction Loss.** $u(p) = 0 \text{ for all } p \in \mathcal{P}$
- **Minimal perimeter.** Multiple ways to connect points, choose the one minimizing

$$\mathrm{per}_{\Omega}(\mathcal{I}) = \int |1_{\mathcal{I}}| = \mathcal{H}^{d-1}(\partial \mathcal{I}) = \mathcal{H}^{d-1}(\mathcal{S})$$

- **Easy way of getting an SDF.** Modify loss functional

$$\mathcal{S} = \{x \in \Omega : \ u(x) = 0\}$$
$$\mathcal{I} = \{x \in \Omega : \ u(x) < 0\}$$
$$\mathcal{O} = \{x \in \Omega : \ u(x) > 0\}$$

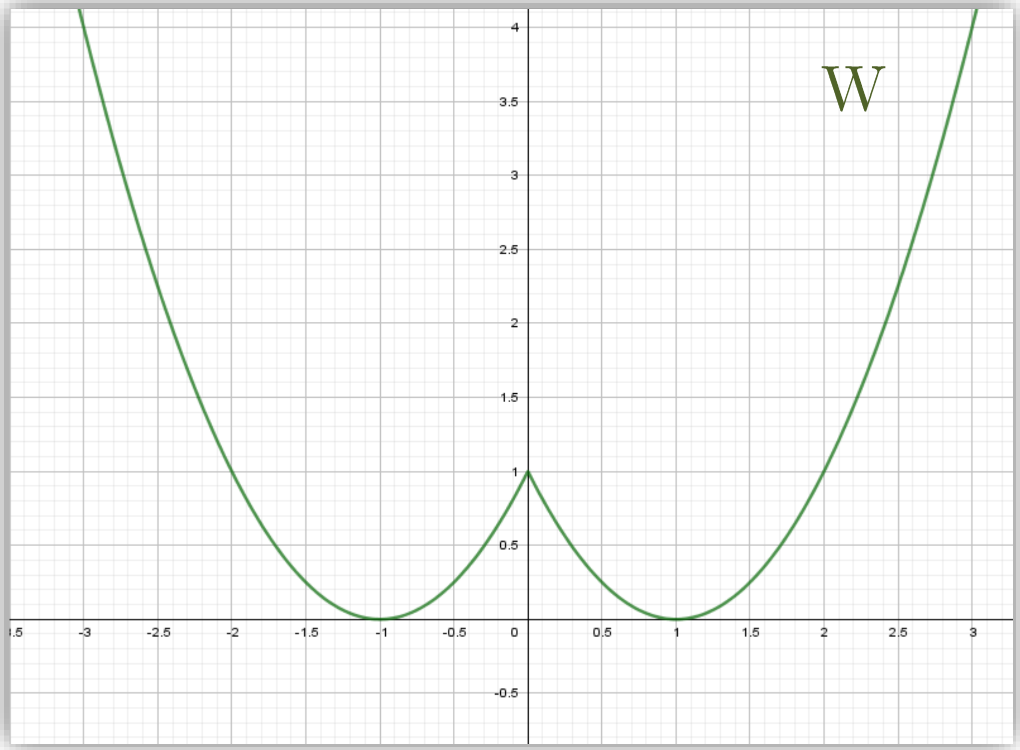
PHASE-Loss | Part 1

- Perimeter difficult to compute
- Use approximation:

$$\mathcal{F}_\varepsilon(u) = \underbrace{\int_\Omega \tfrac{1}{\varepsilon} W(u) + \varepsilon \|\nabla u\|^2 \,\mathrm{d}x}_{\text{PDE}} + \underbrace{\tfrac{\eta}{|\mathcal{P}|} \sum_{p\in\mathcal{P}} \left| \tfrac{1}{|B_\delta|} \int_{B_\delta(p)} u(x)\,\mathrm{d}x \right|}_{\text{Data loss}}$$

Double-Well potential

$$\mathcal{F}_\varepsilon(u) = \int_\Omega \boxed{\tfrac{1}{\varepsilon} W(u)} + \varepsilon \|\nabla u\|^2 \,\mathrm{d}x + \frac{\eta}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \left| \frac{1}{|B_\delta|} \int_{B_\delta(p)} u(x) \,\mathrm{d}x \right|$$



<u>Double-well potential</u>

- $W(s) = 0$ if and only if $s \in \{\pm 1\}$
- $W(s) \to \infty$ for $|s| \to \infty$

- ± 1 are the only global minima
- $\min W(u)$ forces $u(x) \in \{\pm 1\}$
- $x \in \{\mathcal{I}, \mathcal{O}\}$
- proper occupancy

$W(s) = s^2 - 2|s| + 1 \longrightarrow$ non-differntiable in 0, local=global minima

Dirichlet-Regularizer

$$\mathcal{F}_\varepsilon(u) = \underbrace{\int_\Omega \tfrac{1}{\varepsilon} W(u) + \boxed{\varepsilon \|\nabla u\|^2} \,\mathrm{d}x}_{\text{Modica-Mortola}} + \tfrac{\eta}{|\mathcal{P}|} \sum_{p\in\mathcal{P}} \left| \tfrac{1}{|B_\delta|} \int_{B_\delta(p)} u(x)\,\mathrm{d}x \right|$$

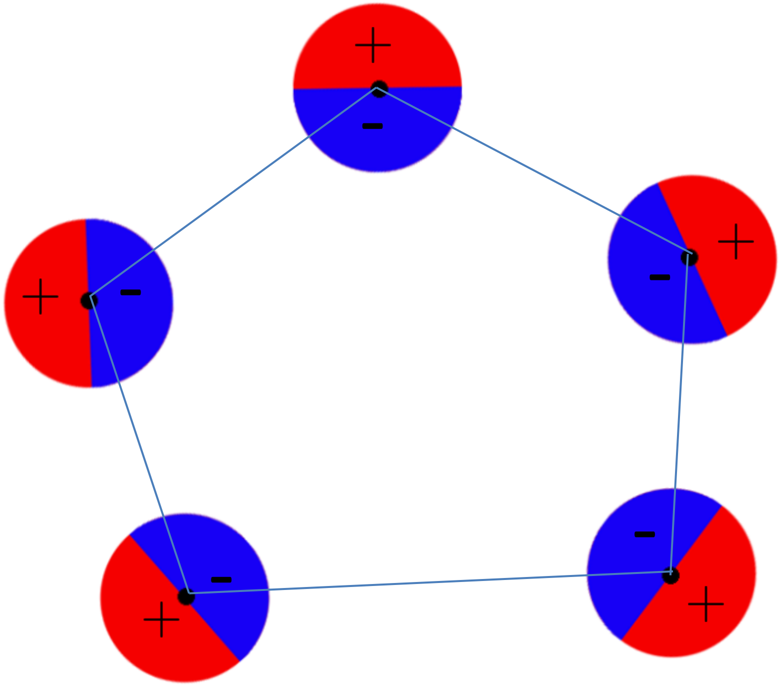
- Similar as regularization in machine learning
- Important for convergence of the functionals

$$\int_\Omega \tfrac{1}{\varepsilon} W(u) + \varepsilon \|\nabla u\|^2 \,\mathrm{d}x \xrightarrow{\Gamma} C\mathrm{per}_\Omega(\mathcal{I})$$

- C depends on double-well
- convergence of minimizers
- minimal perimeter!

Zero-reconstruction loss

$$\mathcal{F}_\varepsilon(u) = \int_\Omega \frac{1}{\varepsilon} W(u) + \varepsilon \|\nabla u\|^2 \, \mathrm{d}x + \boxed{\frac{\eta}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \left| \frac{1}{|B_\delta|} \int_{B_\delta(p)} u(x) \, \mathrm{d}x \right|}$$



- u vanishs along the orignal points
- ZRL-property
- Enforces change of sign

Gamma-convergence

- Can still show convergence

$$\mathcal{F}_\varepsilon(u) = \int_\Omega \tfrac{1}{\varepsilon} W(u) + \varepsilon \|\nabla u\|^2 \,\mathrm{d}x + \tfrac{\eta}{|\mathcal{P}|} \sum_{p\in\mathcal{P}} \left| \tfrac{1}{|B_\delta|} \int_{B_\delta(p)} u(x)\,\mathrm{d}x \right|$$

Theorem

If we choose $\eta = \eta(\varepsilon)$ such that $\lim_{\varepsilon\to 0} \eta = \infty$ and $\lim_{\varepsilon\to 0} \eta\sqrt{\varepsilon} = 0$, then $\mathcal{F}_\varepsilon \xrightarrow{\Gamma} \mathcal{F}_0$

$$\mathcal{F}_0(u) = \begin{cases} C\mathrm{per}_\Omega(\mathcal{I}) & u \in BV \text{ and } \forall p : \int_{B_\delta} u(p) = 0 \\ \infty & else \end{cases}$$

- No information for fixed ε on η
- Problem: ∞ for point cloud sampled from circle

Signed distance function

- find SDF
- $u_\varepsilon = \text{argmin}_u \mathcal{F}_\varepsilon(u)$

LOG-Transform

$$w_\varepsilon(x) := \varepsilon \log(1 - |u_\varepsilon|)\text{sgn}(u_\varepsilon)$$

- Implicit representation

Theorem

In $Q \subset \Omega - \bigcup_p B_\delta(p)$ it holds

$$-\varepsilon \Delta w_\varepsilon + \text{sgn}(w_\varepsilon)(\|\nabla w_\varepsilon\|^2 - 1) = 0$$

- $\longrightarrow \|\nabla w\| = 1$; Eikonal
- w is SDF

PHASE-loss | Part 2

- w is SDF in Q; unclear in $\mathcal{P}$?

$$\boxed{\min_u \mathcal{F}_\varepsilon(u) + \frac{\mu}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} \left|1 - \|\nabla w_\varepsilon(p)\|\right|}$$

- Don't need to calculate the log-transform $\nabla w_\varepsilon(p) = \varepsilon \nabla u_\varepsilon(p) \ \forall p \in \mathcal{P}$

- Eikonal part gives improvement for small PC, irrelevant for dense ones

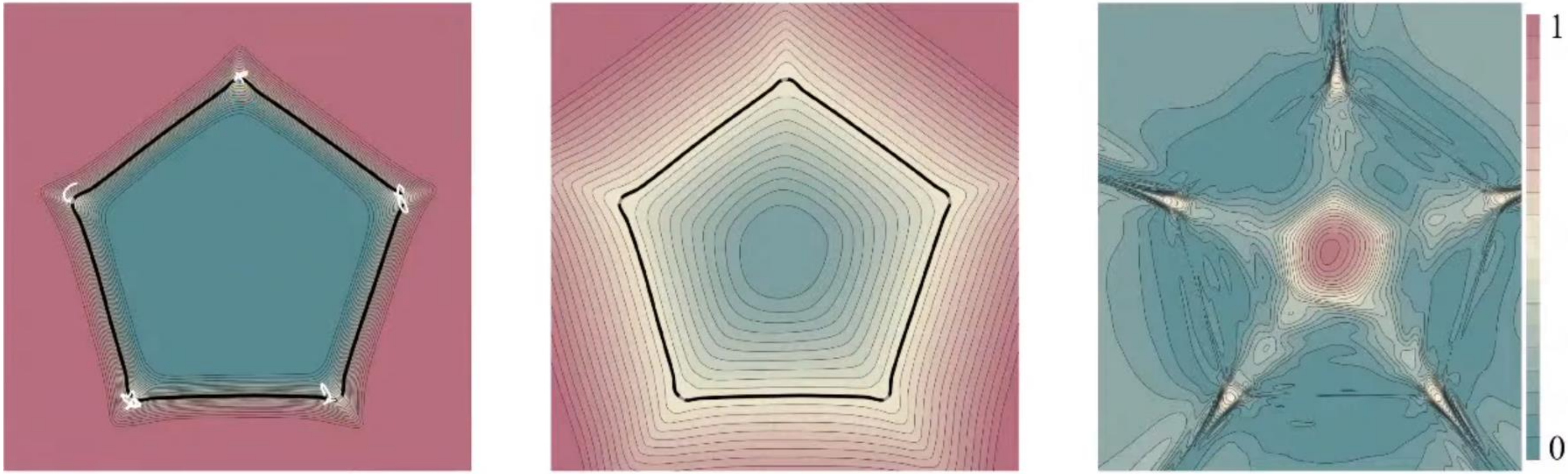
- If we have gradient information

$$\boxed{\min_u \mathcal{F}_\varepsilon(u) + \frac{\mu}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} |n(p) - \nabla w_\varepsilon(p)|}$$

More Eikonal

[*3DGV Seminar: Yaron Lipman --- Unifying Implicit Neural Representations*, *https://www.youtube.com/watch?v=lmjpyIWIsZg&t=425s*, Lipman, 2021]

Reconstruction from 5 points



u_ε w_ε $||\nabla w| - 1|$

- w is smooth, but an SDF is non-differentiable along the Medial axis & $\mathcal{P}$
- SDF and OF by single network

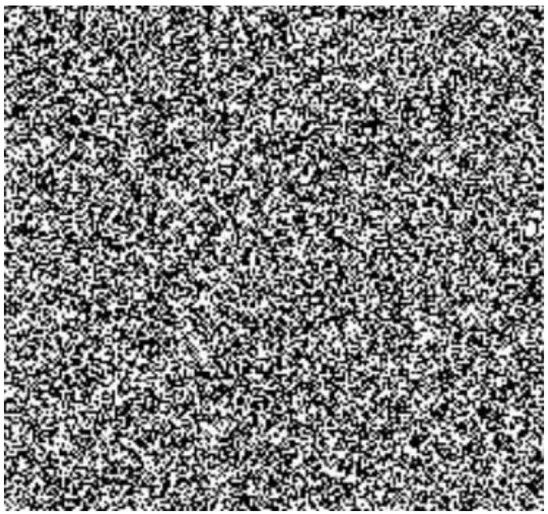
AMBROSIO-TORTORELLI-BASED

Denoising problem

- Consider an image, affiliated with noise

=

+



G U N

How do we denoise the image?

How can we get U from G without knowing N?

Weak Mumford-Shah

Consider $u, g, n : \Omega \to [0, 1]$, with $g = u + n$, reformulate denoising:

Weak Mumford-Shah

$$\min_u \int_\Omega |u - g|^2 + \int_\Omega \|\nabla u\|^2 \,\mathrm{d}x + \mathcal{H}^{d-1}(S_u)$$

- S_u is the jump set of the function u, not continuous
- difficult to compute
- same as perimeter: Approximate

Ambrosio-Tortorelli

$$\mathcal{E}(u) = \int_\Omega |u-g|^2 + \int_\Omega \|\nabla u\|^2 \,\mathrm{d}x + \mathcal{H}^{d-1}(S_u)$$

Ambrosio-Tortorelli

$$\min_{u,v} \mathcal{E}_\varepsilon(u,v) := \int_\Omega |u-g|^2 + \int_\Omega (v(x)^2 + k_\varepsilon)|\nabla u(x)|^2 \,\mathrm{d}x + \int_\Omega \varepsilon |\nabla v(x)|^2 + \frac{(1-v(x))^2}{4\varepsilon} \mathrm{d}x$$

Idea: $v(x) \approx 1_{S_u^c}$, last part approximates jumpset

Theorem

If $k_\varepsilon/\varepsilon \longrightarrow 0$, then $\mathcal{E}_\varepsilon \xrightarrow{\Gamma} \mathcal{E}_0$

$$\mathcal{E}_0(u,v) = \begin{cases} \mathcal{E}(u) & u \in BV(\Omega), v = 1 \ a.e. \\ \infty & else \end{cases}$$

Ambrosio-Tortorelli for surface reconstruction

SR problem via indicator function

$$\mathcal{S} = \{x \in \Omega : 1_{\mathcal{S}^C}(x) = 0\}$$

- In the indicator approach, $S_{1_{\mathcal{S}^C}} = \mathcal{S}$
- AT for jumpsets
- Replace similarity condition with ZRL
- only depends on v

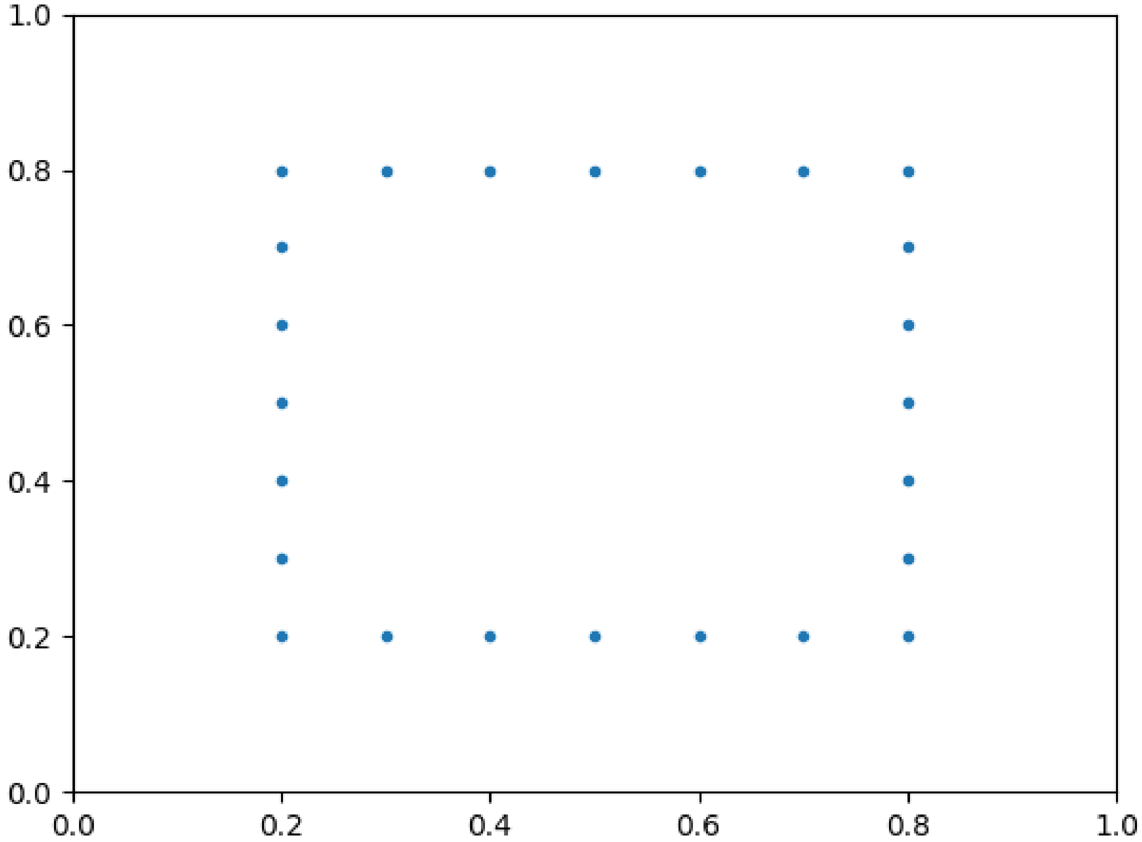
$$\boxed{\min_v \mathcal{E}_\varepsilon(v) := \int_\Omega \frac{(1 - v(x))^2}{4\varepsilon} + \varepsilon \|\nabla v(x)\|^2 + \frac{\eta}{|\mathcal{P}|} \sum_{p \in \mathcal{P}} |v(p)|}$$

- compared to MM replaced double-well with only zero
- no distinction between interior and exterior
- open surfaces
- no perimeter convergence

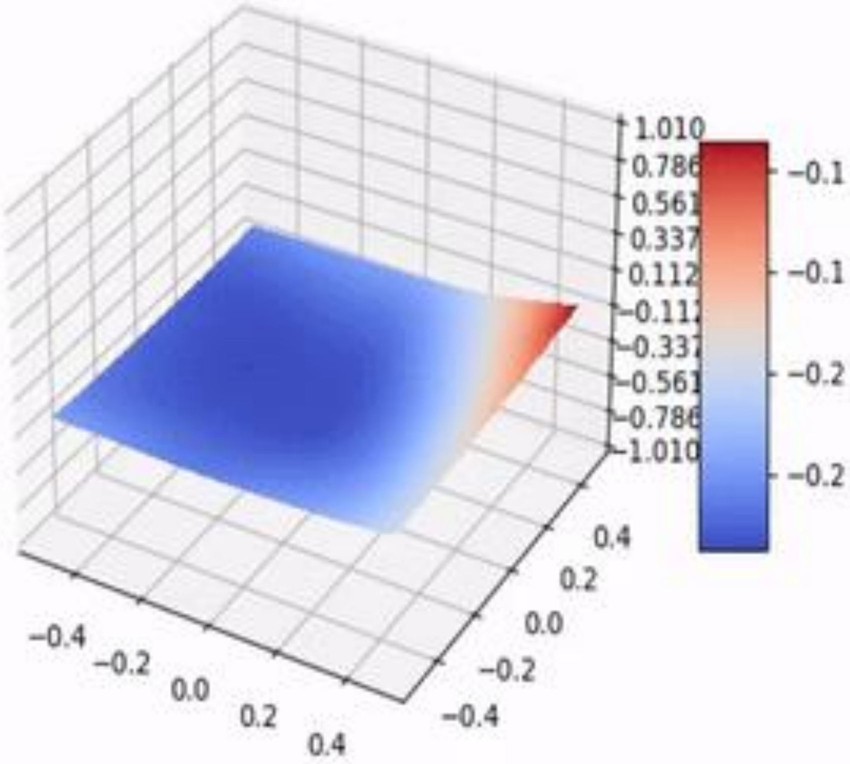
NUMERICAL RESULTS

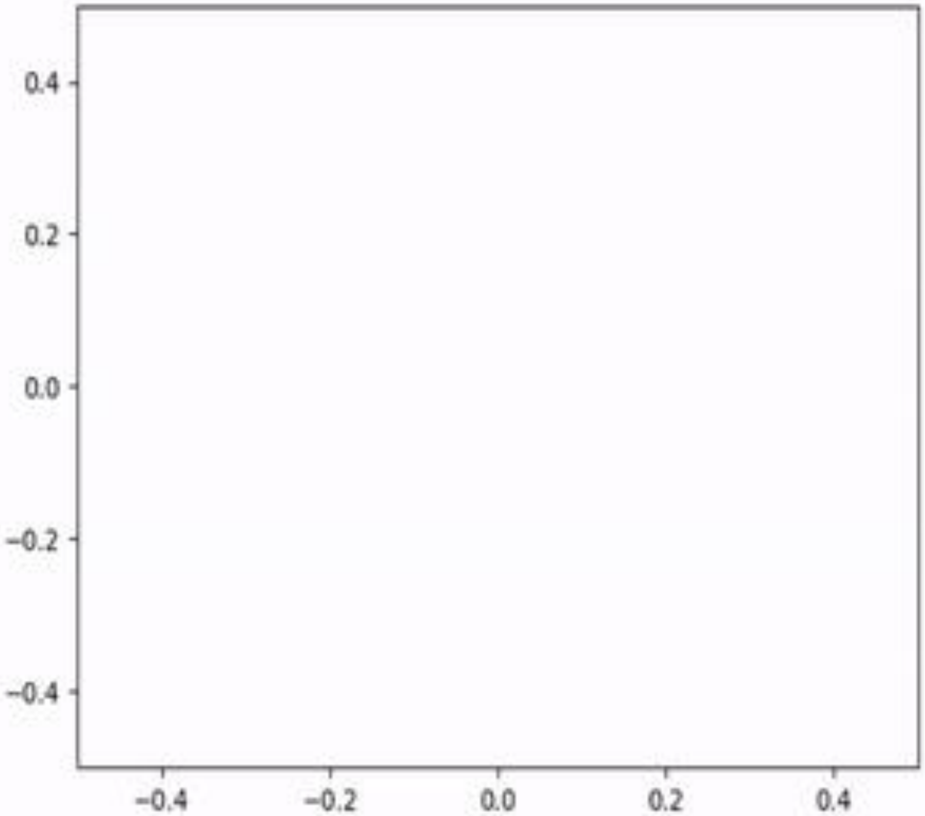
Modica-Mortola-based

- Point Cloud (24) square



- MLP(128)
- ADAM
- 7000 epochs ~ 5 min



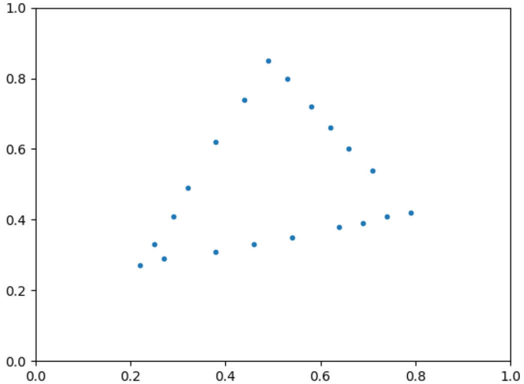


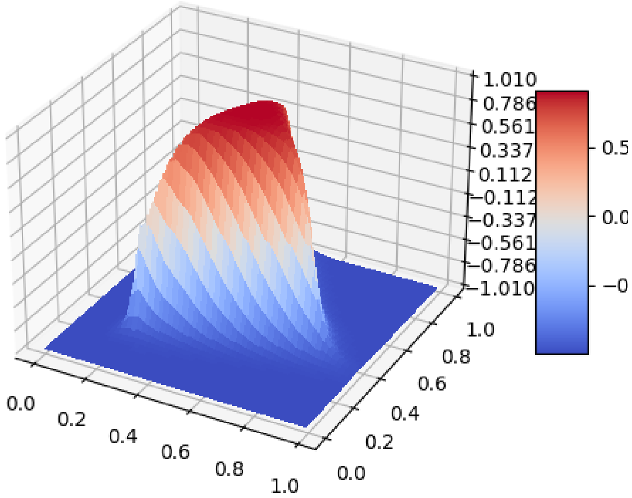
3D Plot of function

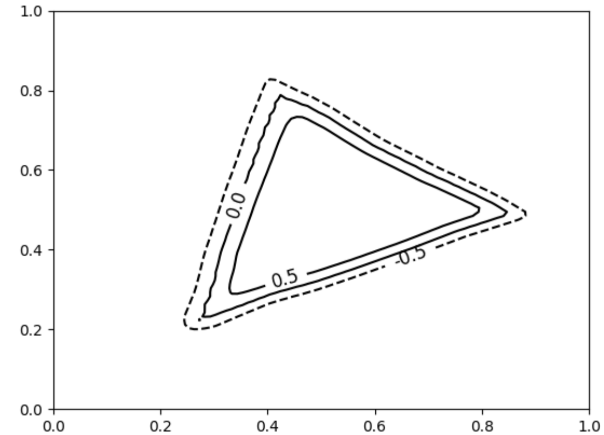
contour plot

$u(p) = 0$, changes sign
level set $\approx$ square

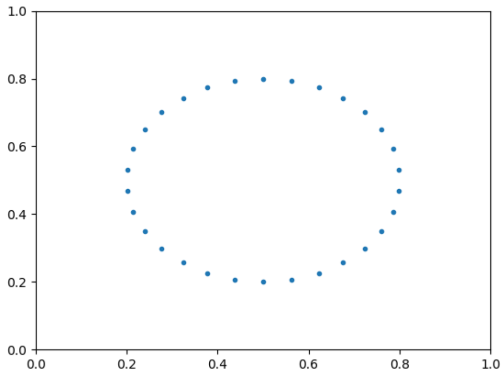
triangle

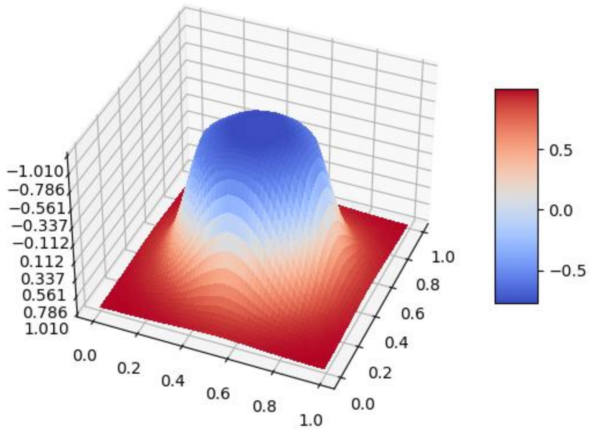


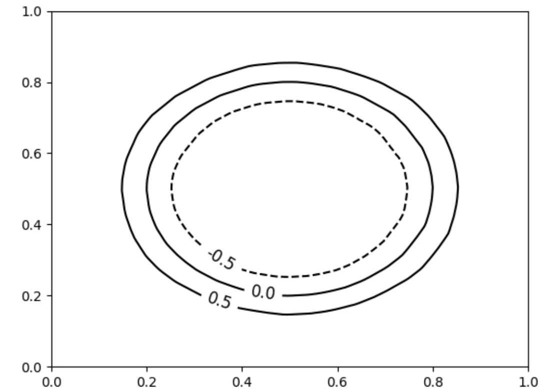




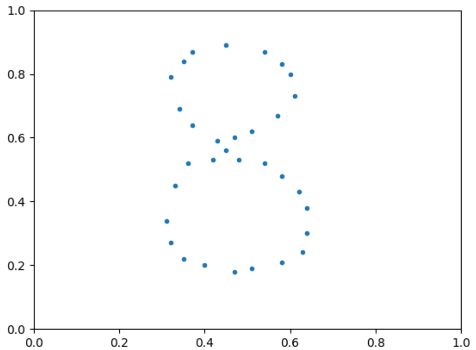
circle

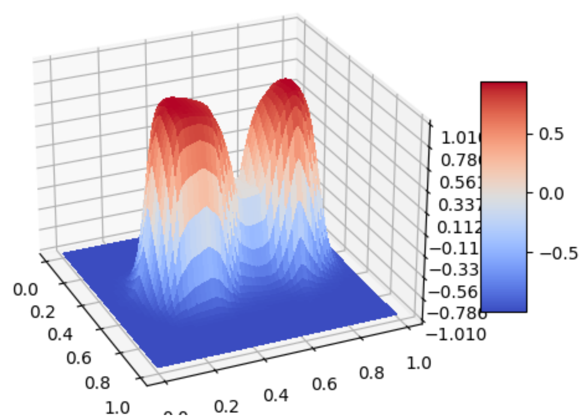


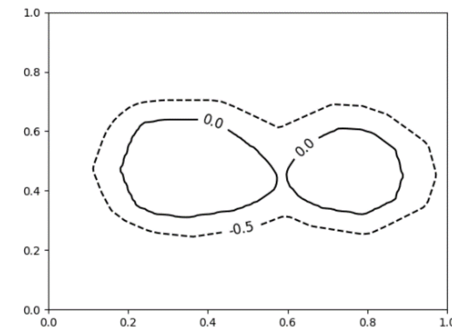




eight

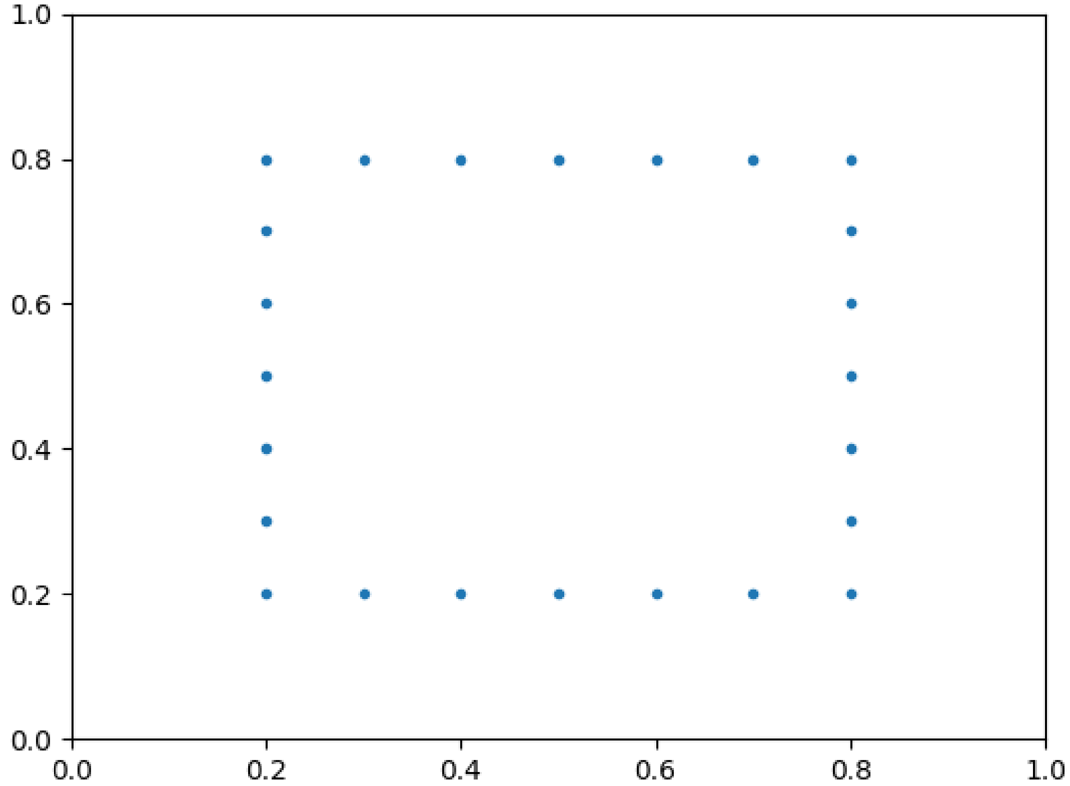




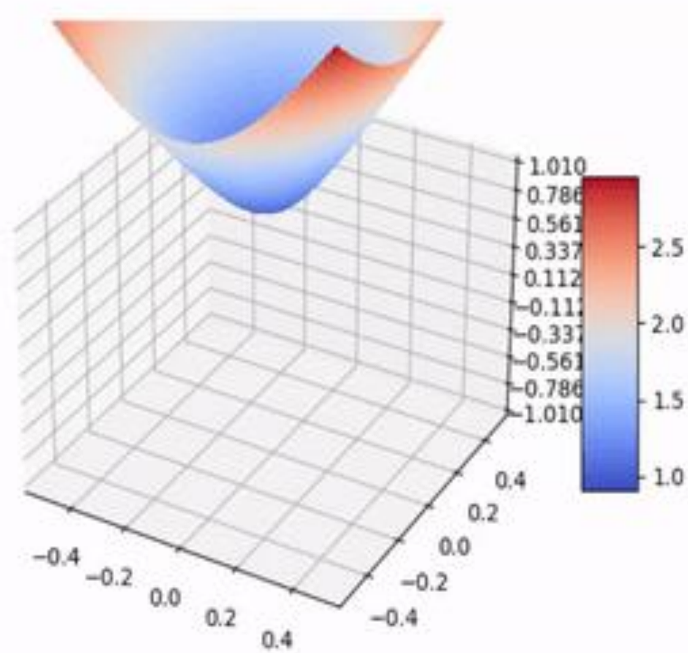


Ambrosio-Tortorelli-based

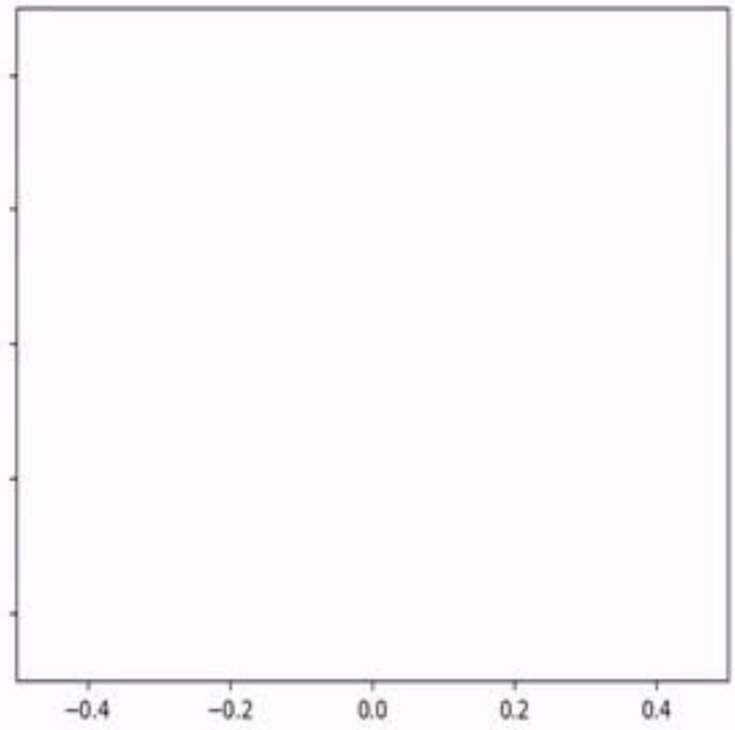
- Point Cloud size 24 from square



Ambrosio-Tortorelli-based



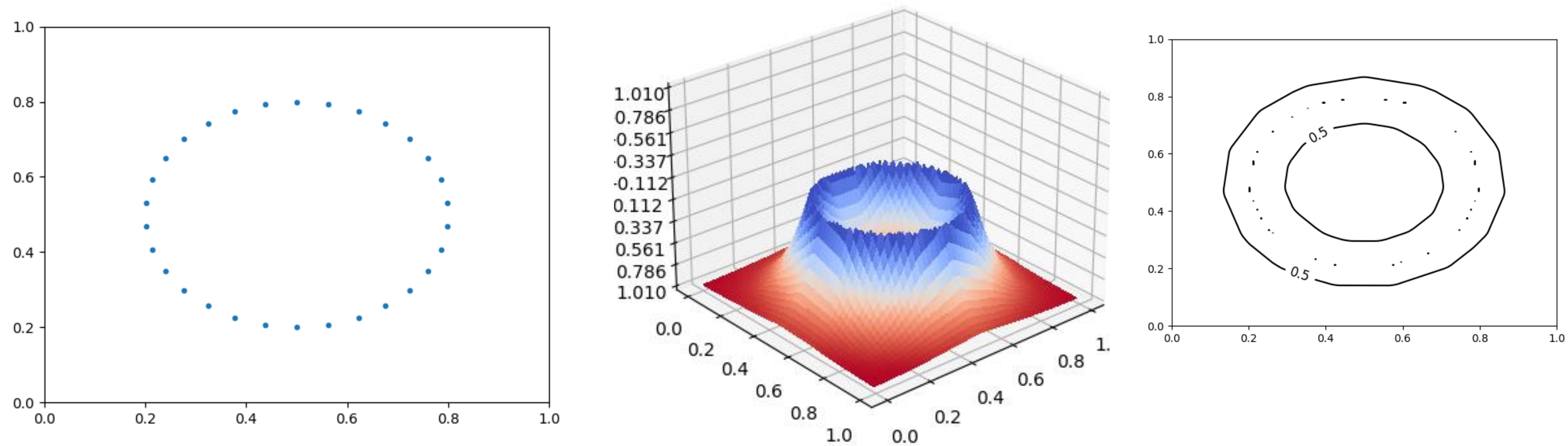
3D Plot of function



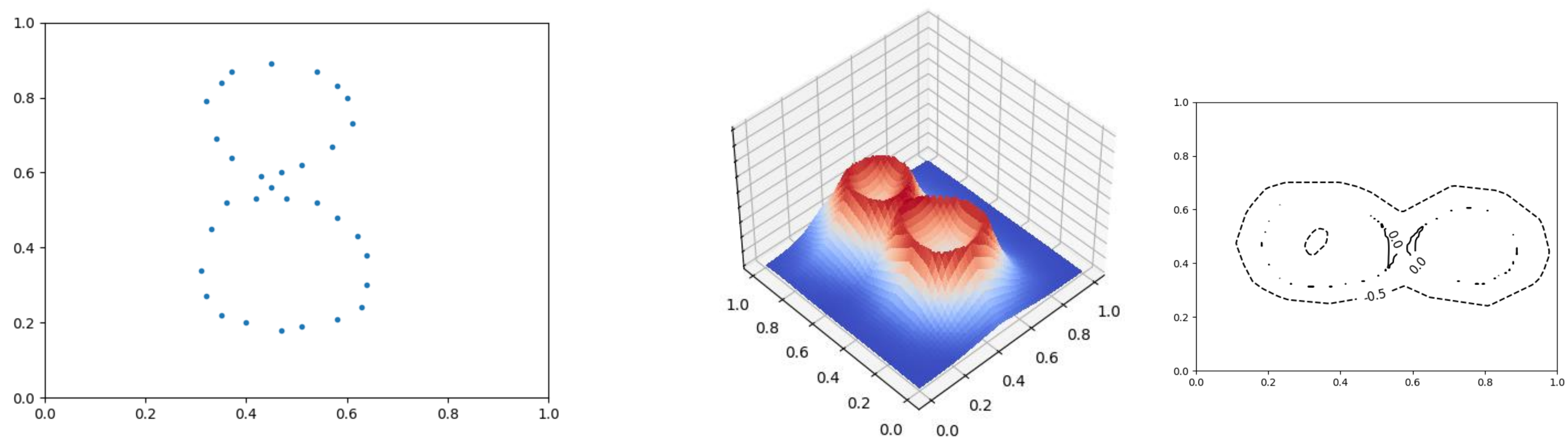
contour plot

- Goes to 0 close to PC, drops back to 1
- no crossing -> no 0-level set

circle

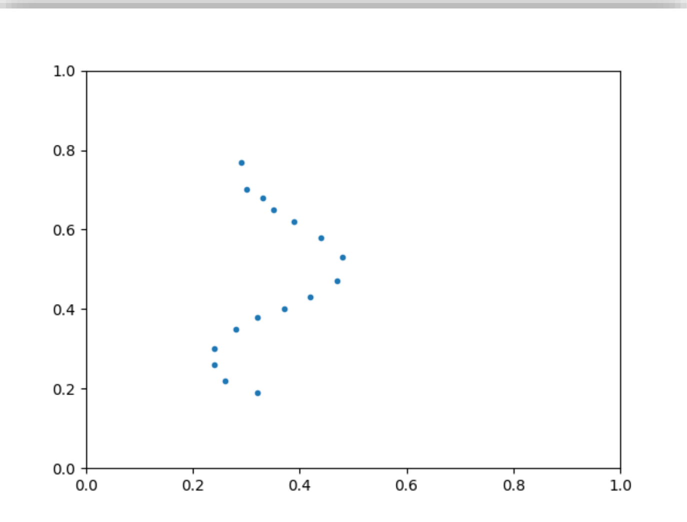


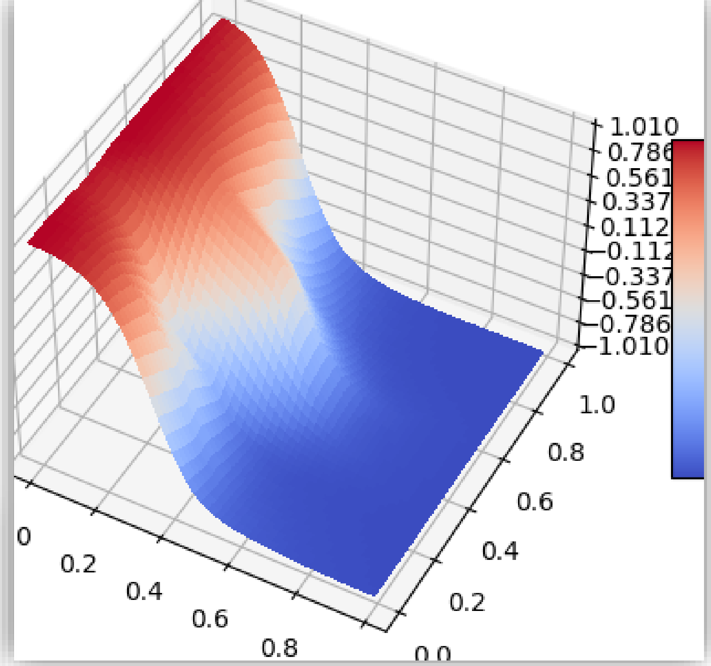
eight



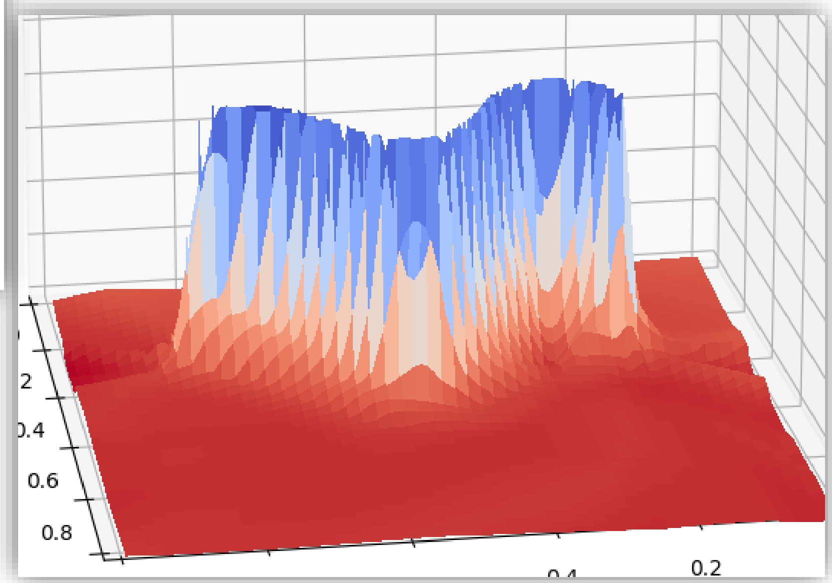
Goes to 0, drops back to 1

- AT no interior exterior
- input: open path





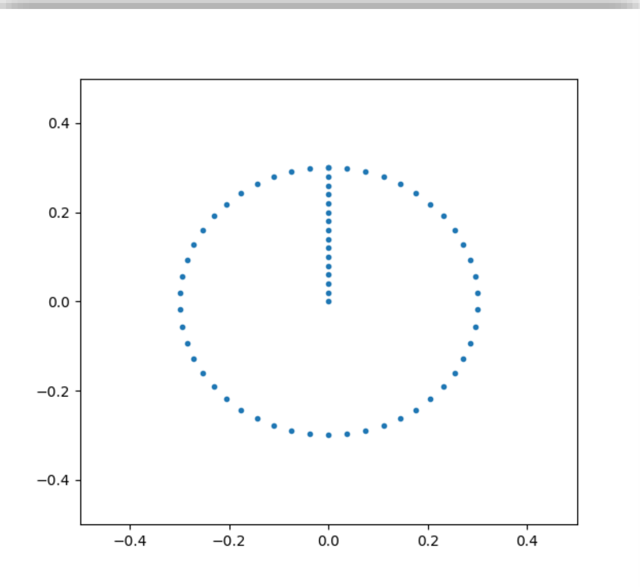
MM:
- Endpoint to boundary
- change sign
- domain left/right

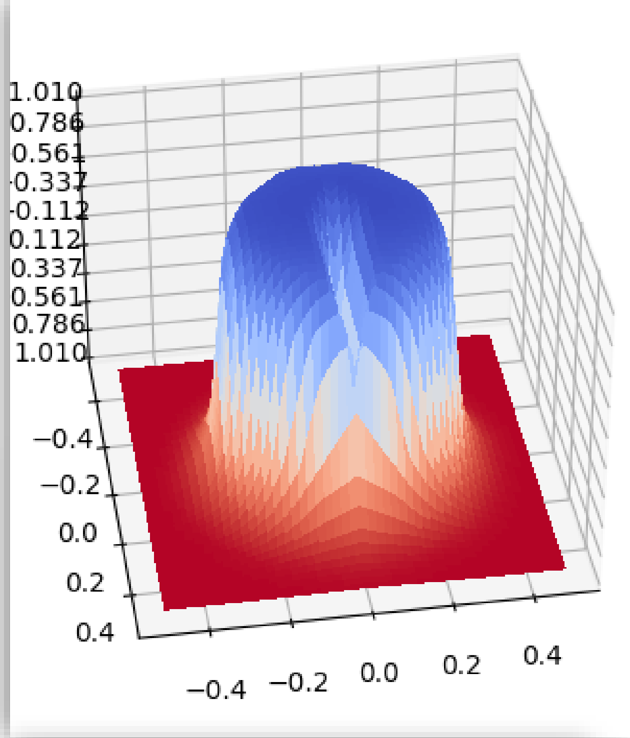


AT:
- no need to drop to 0

Input:

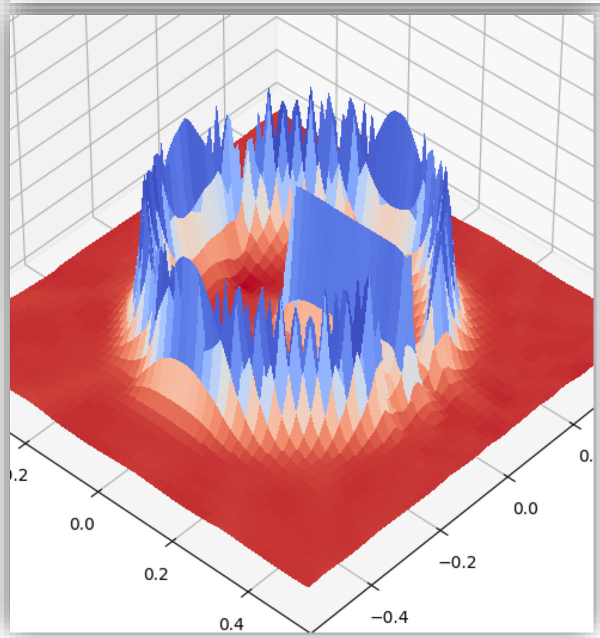
- Circle & line to boundary





MM:

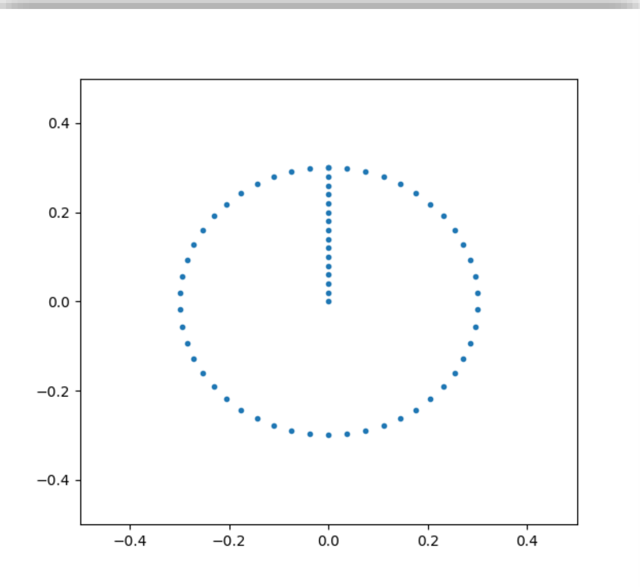
- had Seperation without line
- change inside works against boundary
- Eikonal term

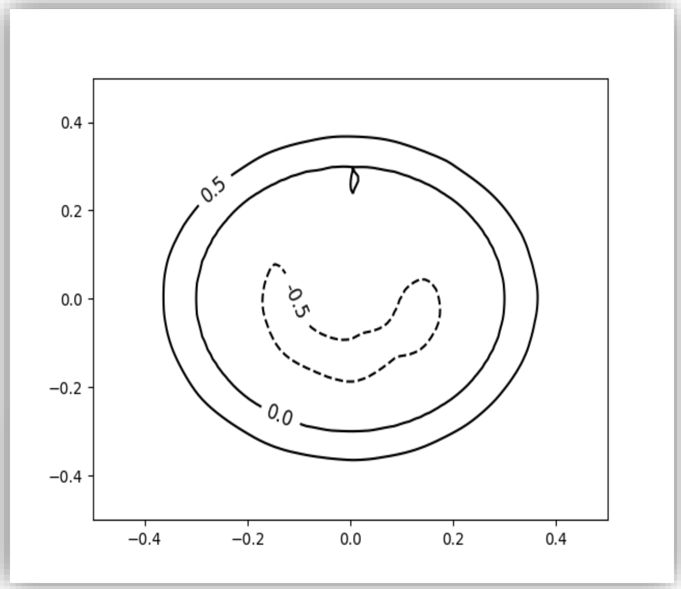


AT:

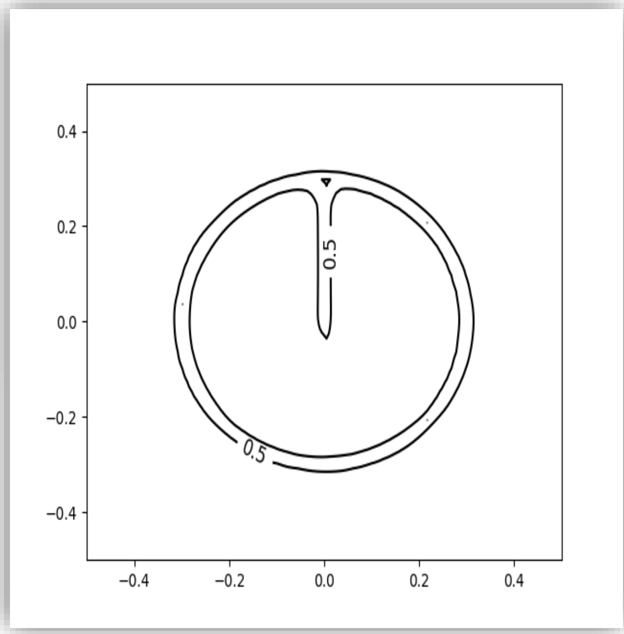
- No separation; no problem
- spikes artefact of MQ

Input:
- Circle & line to boundary





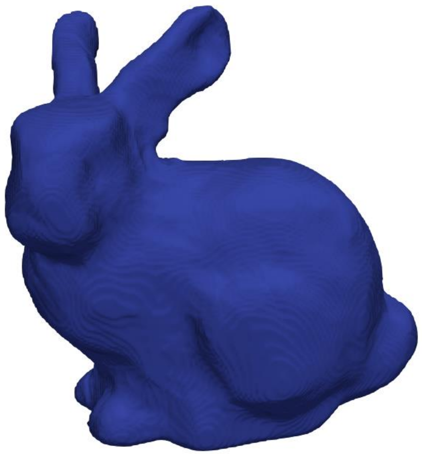
MM:
- 0/0,5 only circle
- -0,5 not accurate

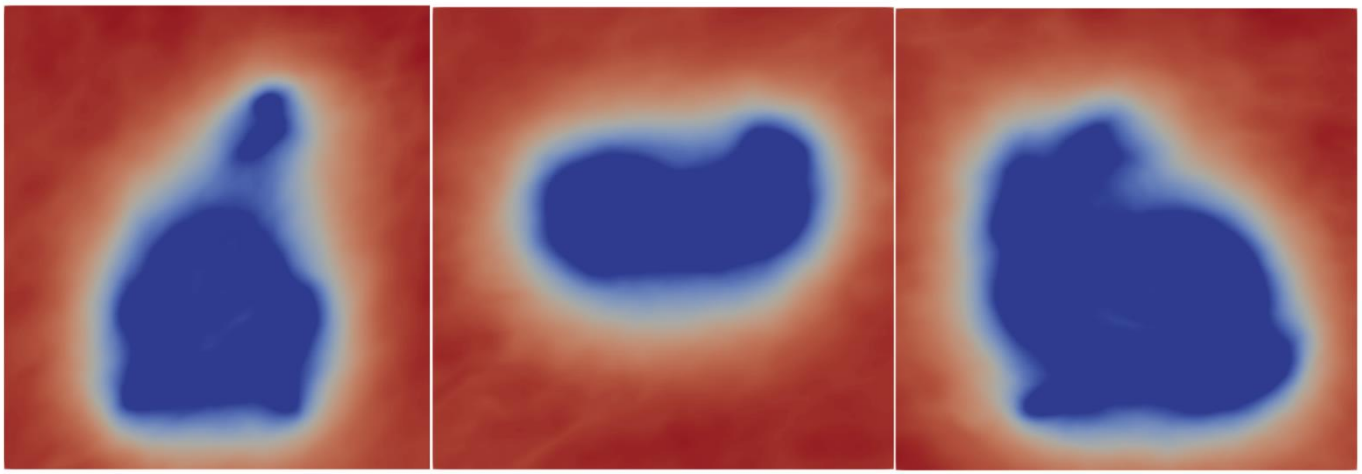


AT:
- see line

3D Examples: MLP(3x512), 50k ~ 30 min.,

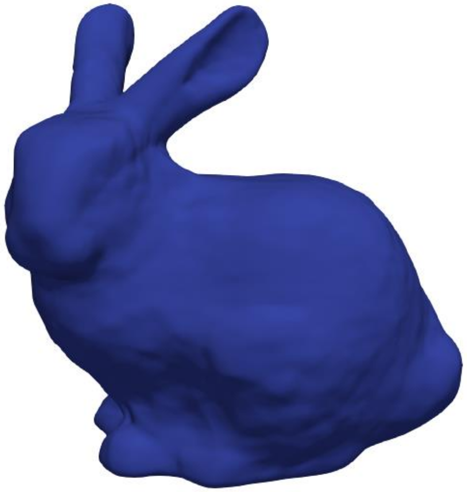
MM

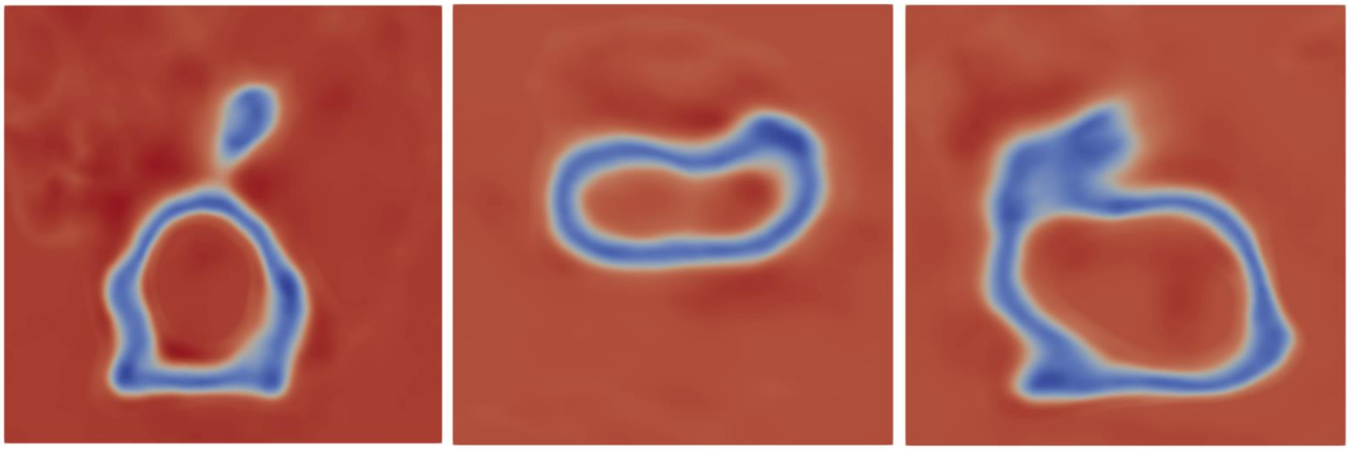




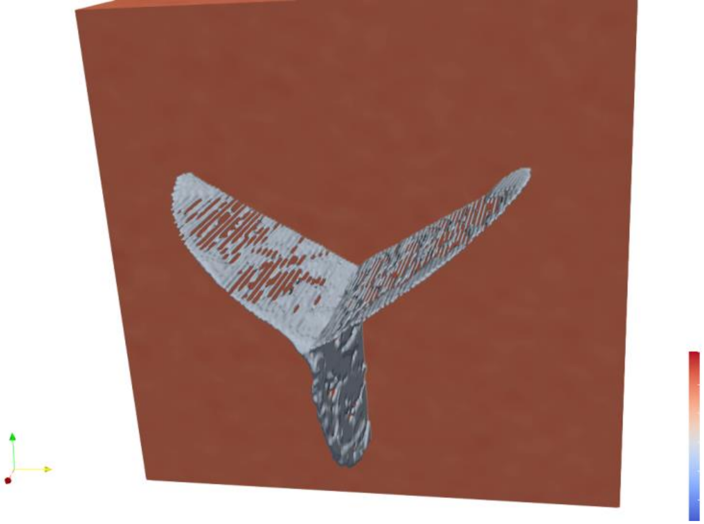
(a) Normal: $(1, 0, 0)$ (b) Normal: $(0, 1, 0)$ (c) Normal: $(0, 0, 1)$

AT



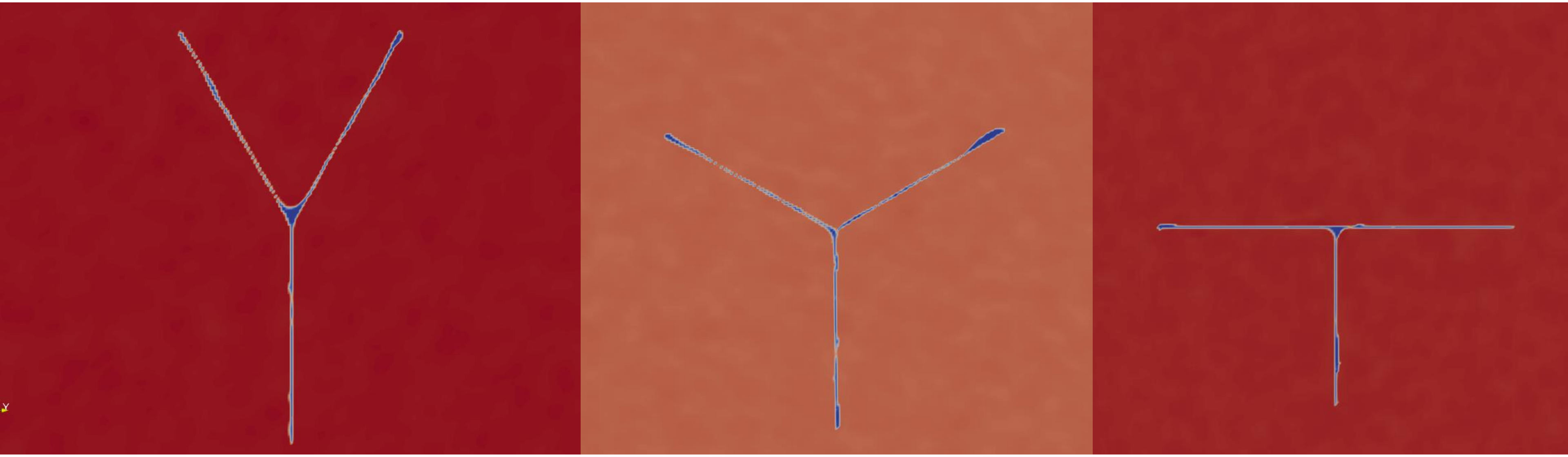


(a) Normal: $(1, 0, 0)$ (b) Normal: $(0, 1, 0)$ (c) Normal: $(0, 0, 1)$



Input: PC from 3 semicircles, different angles, open surface

Cross sections of reconstruction



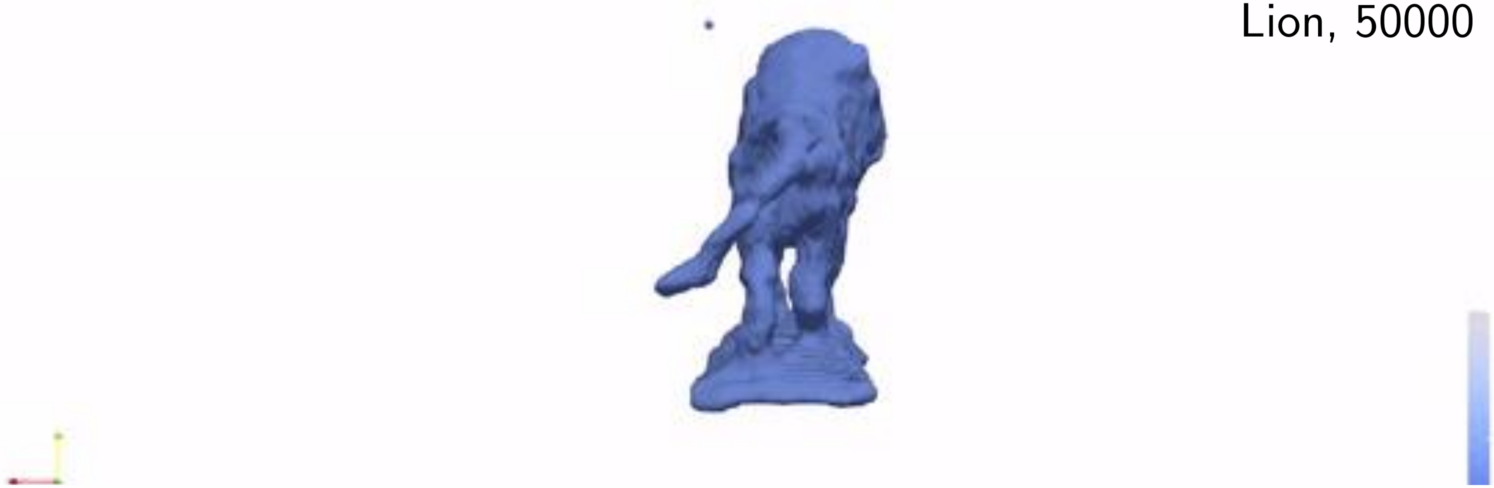
$\frac{5\pi}{6}$ $\frac{2\pi}{3}$ $\frac{\pi}{2}$

- Intersection not filled out, because perimeter is minimized: behaves like minimal perimeter

Armadillo, 40000 points

Skull, 50000 points

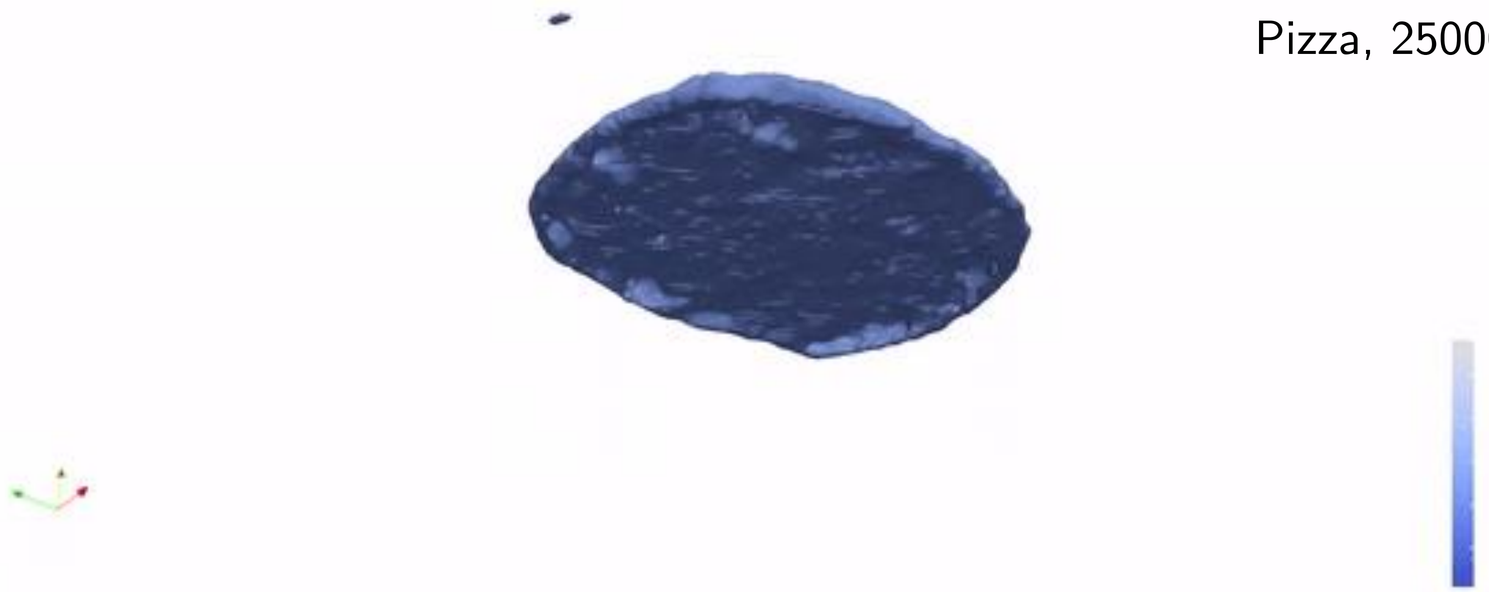
Lion, 50000 points



Koala, 50000 points

Scorpion, 50000 points

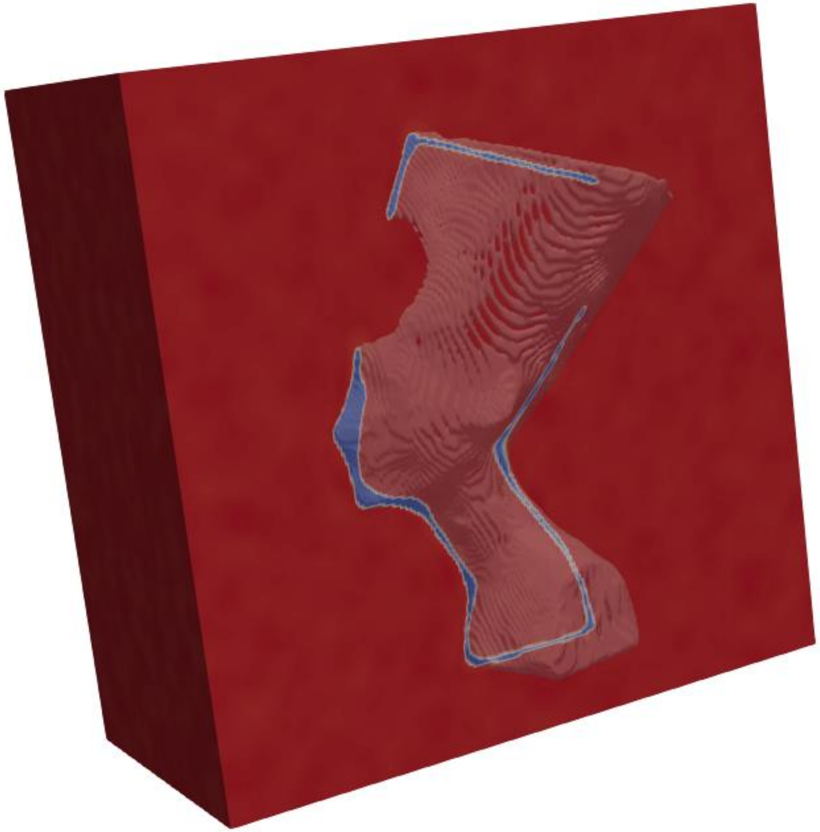
Pizza, 25000 points



Mountain, 25000 points
(Matterhorn)

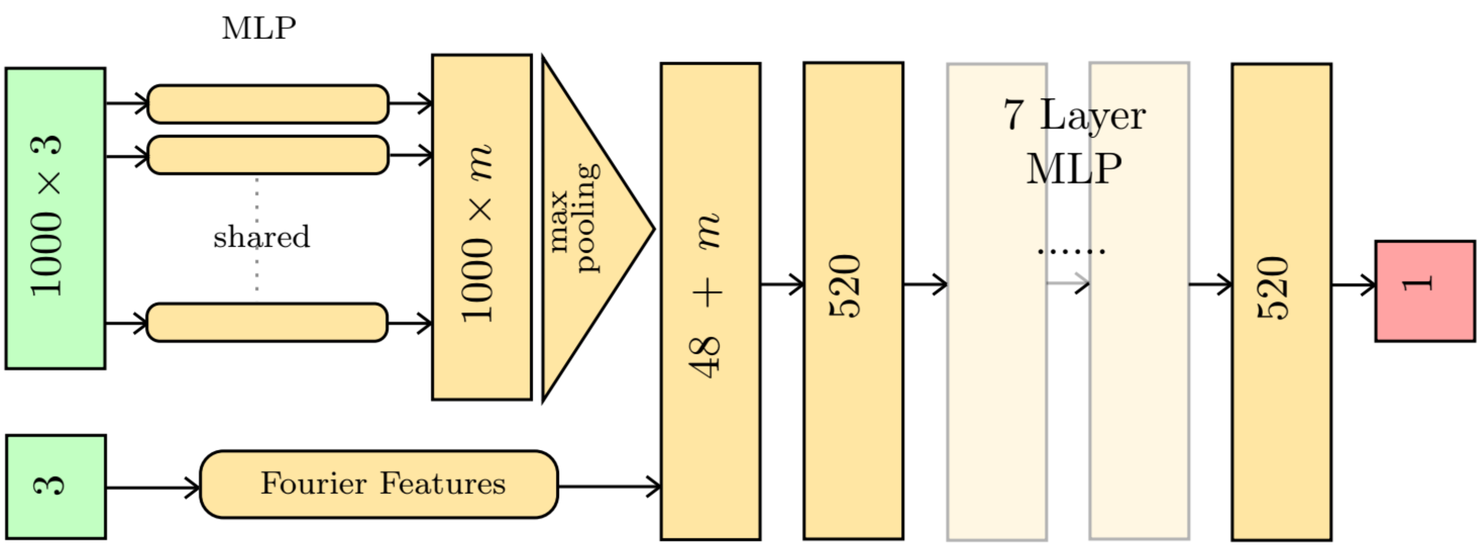
Face, 25000 points

Nofreteti, 25000 points
- Removed points from cylinder



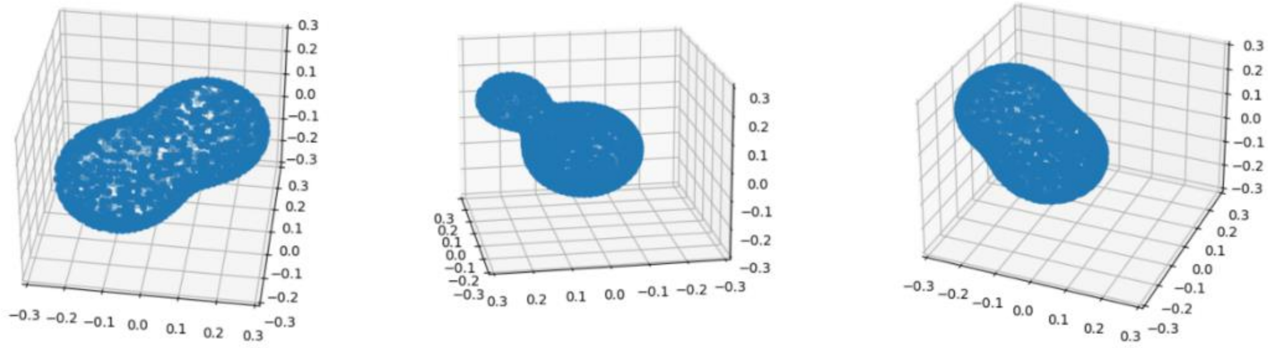
Shape space learning

- Multiple shapes in one network
- network receives object specific input
- extract feature vector from PC using a PointNet Autoencoder
- Only need encoder, train concatenation on sum of Ambrosio-Tortorelli losses

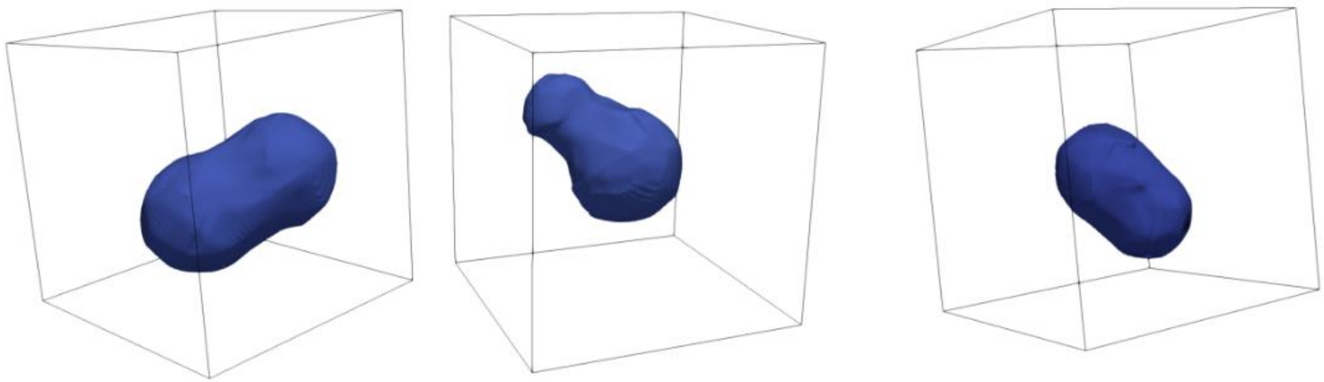


$$\min_{\theta,\psi} \sum_{i=0}^{N} \left(\int_\Omega \frac{1}{4\varepsilon} \left(v_\theta(\xi, E_\psi(\mathcal{P}_i)) - 1 \right)^2 + \varepsilon \left\| \nabla v_\theta(\xi, E_\psi(\mathcal{P}_i)) \right\|^2 \mathrm{d}\xi + \frac{\lambda}{|\mathcal{P}_i|} \sum_{p \in \mathcal{P}_i} \left| v_\theta(p, E_\psi(\mathcal{P}_i)) \right| \right)$$

Experiments for Shape space learning



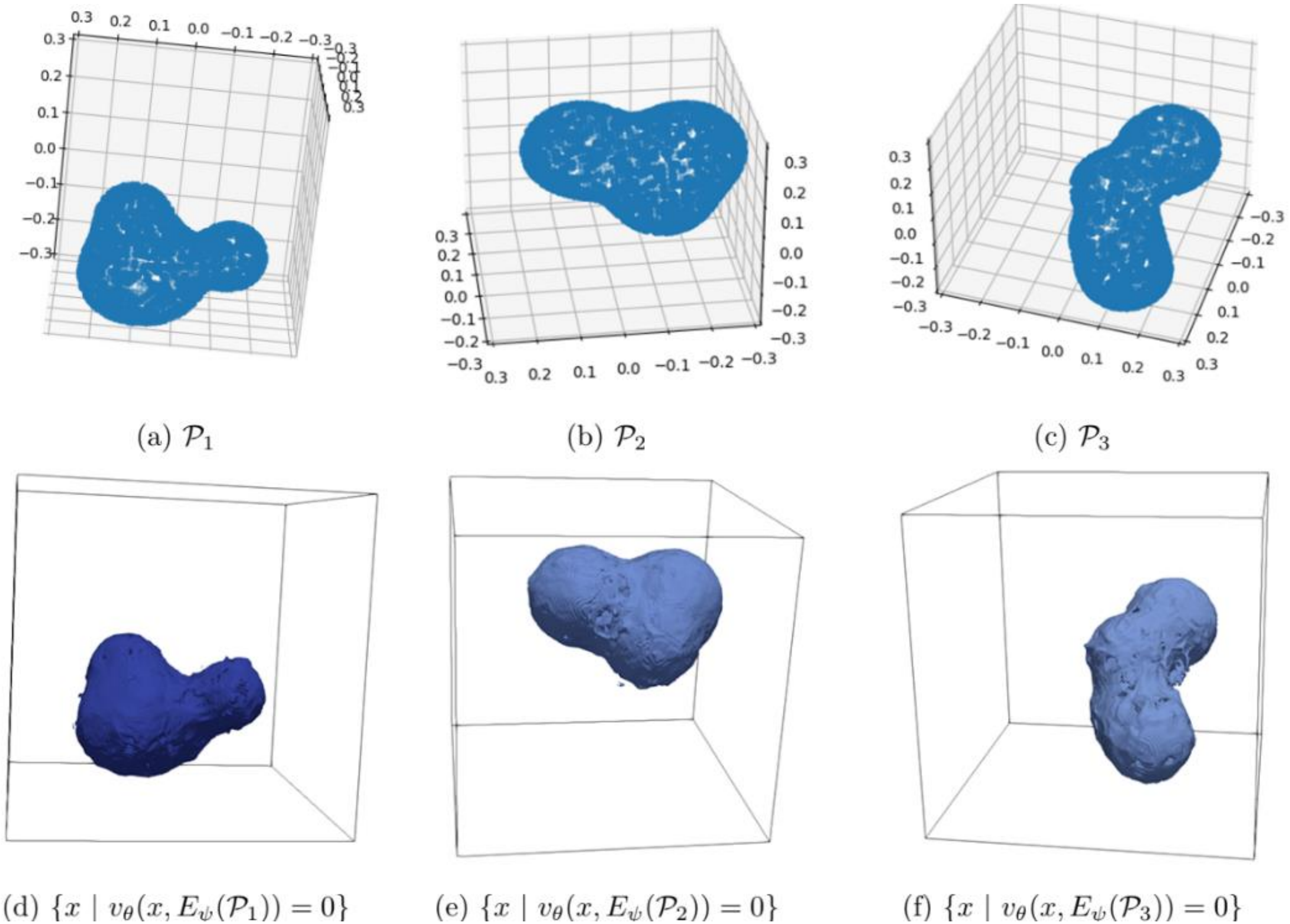
(a) $\mathcal{P}_1$ (b) $\mathcal{P}_2$ (c) $\mathcal{P}_3$



(d) $\{x \mid v_\theta(x, E_\psi(\mathcal{P}_1)) = 0\}$ (e) $\{x \mid v_\theta(x, E_\psi(\mathcal{P}_2)) = 0\}$ (f) $\{x \mid v_\theta(x, E_\psi(\mathcal{P}_3)) = 0\}$

Dataset of Metaballs, 2 spheres, 8-dimensional feature space

Experiments for Shape space learning



(a) $\mathcal{P}_1$ (b) $\mathcal{P}_2$ (c) $\mathcal{P}_3$

(d) $\{x \mid v_\theta(x, E_\psi(\mathcal{P}_1)) = 0\}$ (e) $\{x \mid v_\theta(x, E_\psi(\mathcal{P}_2)) = 0\}$ (f) $\{x \mid v_\theta(x, E_\psi(\mathcal{P}_3)) = 0\}$

Dataset of Metaballs, 4 spheres, 16-dimensional feature space

THANK YOU FOR YOUR ATTENTION!

Fourier-Features

$$\gamma(x) := \left(\cos(2\pi b_1^T x), \quad \sin(2\pi b_1^T x), \quad \ldots \quad \cos(2\pi b_M^T x), \quad \sin(2\pi b_M^T x)\right)^T.$$

$b_i \approx \mathcal{N}(0, \sigma^2)$